

```

def canEat(candiesCount, queries):
    n = len(candiesCount)

    # حساب العدد التراكمي للحلول حتى النوع i
    prefixSums = [0] * (n + 1)
    for i in range(n):
        prefixSums[i + 1] = prefixSums[i] +
candiesCount[i]

    answer = []
    for favoriteType, favoriteDay, dailyCap in
queries:
        # الحد الأدنى والأقصى لعدد الحلوى الذي يمكن
        تناولها حتى اليوم المفضل
        minCandiesNeeded = favoriteDay + 1
        maxCandiesPossible = (favoriteDay + 1) *
dailyCap

        # عدد الحلوى من النوع المطلوب حتى الآن
        candiesBeforeFavoriteType =
prefixSums[favoriteType]
        candiesWithFavoriteType =
prefixSums[favoriteType + 1]

        # التحقق مما إذا كان بإمكاننا تناول الحلوى في
اليوم المطلوب
        canEatOnFavoriteDay = maxCandiesPossible >
candiesBeforeFavoriteType and minCandiesNeeded <=
candiesWithFavoriteType

        answer.append(canEatOnFavoriteDay)

    return answer

```

```

# مثال 1
candiesCount = [7, 4, 5, 3, 8]
queries = [[0, 2, 2], [4, 2, 4], [2, 13, 1000000000]]
print(canEat(candiesCount, queries)) # Output:
[True, False, True]

```

```

# مثال 2
candiesCount = [5, 2, 6, 4, 1]
queries = [[3, 1, 2], [4, 10, 3], [3, 10, 100], [4,
100, 30], [1, 3, 1]]
print(canEat(candiesCount, queries)) # Output:
[False, True, True, False, False]

```

```

def canEat(candiesCount, queries):
    n = len(candiesCount)

    i حساب العدد التراكمي للحلوى حتى النوع #
    prefixSums = [0] * (n + 1)
    for i in range(n):
        prefixSums[i + 1] = prefixSums[i] +
candiesCount[i]

    answer = []
    for favoriteType, favoriteDay, dailyCap in
queries:
        الحد الأدنى والأقصى لعدد الحلوى الذي يمكن
        تناوله حتى اليوم المفضل
        minCandiesNeeded = favoriteDay + 1
        maxCandiesPossible = (favoriteDay + 1) *
dailyCap

        عدد الحلوى من النوع المطلوب حتى الآن
        candiesBeforeFavoriteType =
prefixSums[favoriteType]
        candiesWithFavoriteType =
prefixSums[favoriteType + 1]

        التحقق مما إذا كان بإمكاننا تناول الحلوى في
        اليوم المطلوب
        canEatOnFavoriteDay = maxCandiesPossible >
candiesBeforeFavoriteType and minCandiesNeeded <=
candiesWithFavoriteType

        answer.append(canEatOnFavoriteDay)

    return answer

```

مثال 1

```

candiesCount = [7, 4, 5, 3, 8]
queries = [[0, 2, 2], [4, 2, 4], [2, 13, 1000000000]]
print(canEat(candiesCount, queries)) # Output:
[True, False, True]

```

مثال 2

```

candiesCount = [5, 2, 6, 4, 1]
queries = [[3, 1, 2], [4, 10, 3], [3, 10, 100], [4,
100, 30], [1, 3, 1]]
print(canEat(candiesCount, queries)) # Output:
[False, True, True, False, False]

```